computation

MDPI

Article

Criminal Intention Detection at Early Stages of Shoplifting Cases by Using 3D Convolutional Neural Networks

Guillermo A. Martínez-Mascorro [1], José R. Abreu-Pederzini [1], José C. Ortiz-Bayliss [1,*], Angel Garcia-Collantes [2] and Hugo Terashima-Marín [1]

[1] School of Engineering and Sciences, Tecnologico de Monterrey, Monterrey 64849, Mexico; a00824126@itesm.mx (G.A.M.-M.); a00793921@itesm.mx (J.R.A.-P.); terashima@tec.mx (H.T.-M.)
[2] Department of Criminology, Universidad a Distancia de Madrid, 28400 Madrid, Spain; angel.garcia.c@udima.es
* Correspondence: jcobayliss@tec.mx

Abstract: Crime generates significant losses, both human and economic. Every year, billions of dollars are lost due to attacks, crimes, and scams. Surveillance video camera networks generate vast amounts of data, and the surveillance staff cannot process all the information in real-time. Human sight has critical limitations. Among those limitations, visual focus is one of the most critical when dealing with surveillance. For example, in a surveillance room, a crime can occur in a different screen segment or on a distinct monitor, and the surveillance staff may overlook it. Our proposal focuses on shoplifting crimes by analyzing situations that an average person will consider as typical conditions, but may eventually lead to a crime. While other approaches identify the crime itself, we instead model suspicious behavior—the one that may occur before the build-up phase of a crime—by detecting precise segments of a video with a high probability of containing a shoplifting crime. By doing so, we provide the staff with more opportunities to act and prevent crime. We implemented a 3DCNN model as a video feature extractor and tested its performance on a dataset composed of daily action and shoplifting samples. The results are encouraging as the model correctly classifies suspicious behavior in most of the scenarios where it was tested. For example, when classifying suspicious behavior, the best model generated in this work obtains precision and recall values of 0.8571 and 1 in one of the test scenarios, respectively.

Keywords: 3D convolutional neural networks; crime prevention; pre-crime behavior method; shoplifting; suspicious behavior

Citation: Martínez-Mascorro, G.A.; Abreu-Pederzini, J.R.; Ortiz-Bayliss, J.C.; Garcia-Collantes, A.; Terashima-Marín, H. Criminal Intention Detection at Early Stages of Shoplifting Cases by Using 3D Convolutional Neural Networks. *Computation* **2021**, *9*, 24. https://doi.org/10.3390/computation9020024

Received: 12 December 2020
Accepted: 12 February 2021
Published: 23 February 2021

1. Introduction

According to the 2020 National Retail Security Survey (NRSS) [1], inventory shrink—a loss of inventory related to theft, shoplifting, error, or fraud—had an impact of $61.7 billion in 2019 on the U.S. retail economy. Many scams occur every day, from distractions and bar code switching to booster bags and fake weight strategies, and there is no human power to watch every one of these cases. The surveillance context is overwhelmed. Vigilance camera networks generate vast amounts of video screens, and the surveillance staff cannot process all the available information as fast as needed. The more recording devices become available, the more complex the task of monitoring such devices becomes.

Real-time analysis of surveillance cameras has become an exhaustive task due to human limitations. The primary human limitation is the Visual Focus of Attention (VFOA) [2]. The human gaze can only concentrate on one specific point at once. Although there are large screens and high-resolution cameras, a person can only pay attention to a small segment of the image at a time. Optical focus is a significant human-related disadvantage in the surveillance context. A crime can occur in a different screen segment or on a different monitor, and the staff may not notice it. Other significant difficulties may be related to attention, boredom, distractions, lack of experience, among others [3,4].

Defining what can be considered suspicious behavior is usually tricky, even for psychologists. In this work, the mentioned behavior is related to the commission of a crime, but it does not imply its realization (Figure 1). For this research, we define suspicious behavior as a series of actions that happen before a crime occurs. In this context, our proposal focuses on shoplifting crime scenarios, particularly before the build-up phase situations that an average person may consider as typical conditions. Shoplifting crimes usually take place in supermarkets, malls, retail stores, and other similar businesses. Many of the models for addressing this problem need the suspect to commit a crime to detect it. Examples of such models include face detection of previous offenders [5,6] and object analysis in fitting rooms [7]. In this work, we propose an approach to support the monitoring staff to focus on specific areas of screens where crime is more likely to happen. While existing models identify the crime itself, we model suspicious behavior as a way to anticipate a potential crime. In other words, we identify behaviors that usually take place before a shoplifting crime occurs. Then, the system can label a video: as containing suspicious or normal behavior. By detecting situations in a video that may indicate that suspicious behavior is present, the system indicates that a crime is likely to happen soon. The former gives the surveillance staff more opportunities to act, prevent, or even respond to such a crime. In the end, it is the security personnel who will decide how to proceed in each situation.

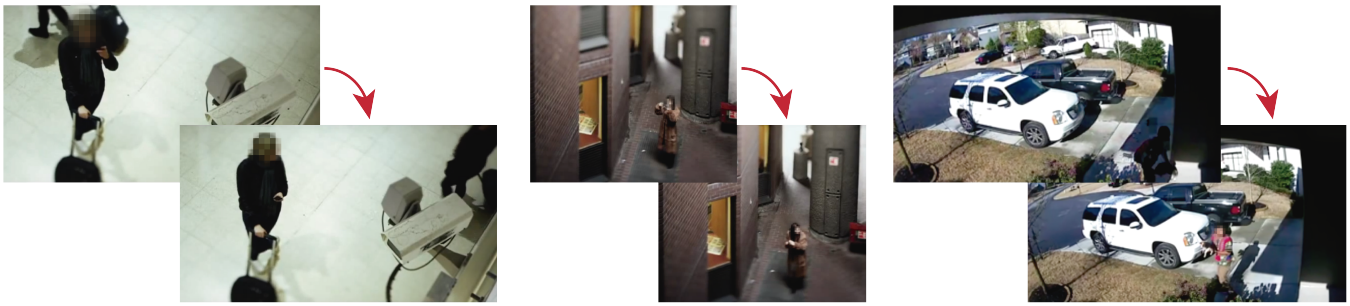


Figure 1. Different situations may be recorded by surveillance cameras. Suspicious behavior is not the crime itself. However, particular situations will make us distrust a person if we consider their behavior to be "suspicious".

Overall, we propose a method to extract segments from videos that feed a model based on a 3D Convolutional Neural Network (3DCNN) for classifying behavior (as normal or suspicious). Once we train the model with such segments, it accurately classifies the behavior on a video dataset composed of daily action samples and shoplifting samples. Our results suggest that the proposed approach has applications in crime prevention in shoplifting cases.

As a summary, this work contributes to the literature mainly in three aspects.

- It describes a methodology, the PCB method, to unify the processing and division of criminal video samples into useful segments that can later be used for feeding a Deep Learning (DL) model.
- It represents the first implementation of a 3DCNN architecture to detect criminal intentions before an offender shows suspicious behavior.
- It provides a set of experiments to validate the results, confirming that the proposed approach is suitable for such a challenging task: to detect criminal intention even before the suspect begins to behave suspiciously.

The remainder of this document is organized as follows. In Section 2, we review various approaches that range from psychology to deep learning, to tackle behavior detection. Section 3 presents the methodology followed to extract the relevant video segments used as input for our model, the PCB method, and the DL model architecture. The experiments, results, and their discussion are presented in Section 4. Finally, Section 5 presents the conclusions and future works derived from this investigation.

2. Background and Related Work

A surveillance environment must satisfy a particular set of requirements. Those requirements have promoted the creation of specialized tools, both on equipment and software, to support the surveillance task. The most common approaches include motion detection [8,9], face recognition [5,6,10,11], tracking [12–14], loitering detection [15], abandoned luggage detection [16], crowd behavior [17–19], and abnormal behavior [20,21]. Prevention and reaction are two primary aims in the surveillance context. Prevention requires forestalling and deterring crime execution. The monitoring staff must remain alert, watch as much as possible, and alert the ground personnel. Reaction, on the other hand, involves protocols and measures to respond to a specific event. The security teams take action only after the crime or event has taken place.

Most security support approaches focus on crime occurrence. Tsushita and Zin presented a snatching-detection algorithm, which performs background subtraction and pedestrian tracking to make a decision [22]. Their approach divides the frame into eight areas and searches for a speed shift in one tracked person. Unfortunately, Tsushita and Zin's algorithm can only alert when a person has already lost their belongings. Ullah et al. proposed a violence detection framework combining a trained MobileNet-SSD model [23] for person detection and a C3D model [24]. Besides, they optimize the trained model with the OPENVINO toolkit [25]. They test their model with three different violence datasets: violent crowd [26], violence in movies [27], and hockey fight [27]. Sultani et al. presented a real-world anomaly detection approach, training 13 anomalies, such as burglary, fighting, shooting, and vandalism [28]. They use a 3DCNN for feature extraction and label the samples into two categories: normal and anomalous. Their model includes a ranking loss function and trains a fully connected neural network for decision-making. In a similar context, Nassarudin et al. [29] presented a deep anomaly detection approach. They implemented a bilateral background subtraction, use the pretrained C3D model [24] for feature extraction, and attached a fully connected network to perform regression. Using the UCF-Crime dataset [30], they trained their model on 11 complete classes and tested their results on "robbery", "fighting", and "road accidents". Ishikawa and Zin proposed a system to detect loitering people [31]. Their system combines grid-based analysis, direction-based analysis, distance-based analysis, acceleration based analysis, and a decision-fusion stage of the people shown in the video to make a decision. Afra and Alhajj [32] proposed a surveillance system that performs face detection and, according to the response, raises the alarm or tries to evaluate the suspect social media. Through security cameras, they collected images and processed them for face detection. They implemented the MobileNet-v1 [23] architecture and trained on the WIDER face dataset [33]. After the face location, they performed a face recognition by implementing two feature extraction techniques: OpenFace [34] and Inception-Resnet-v1 [35], trained on MS-Celeb-1M [36].

Convolutional Neural Networks (CNN) have shown a remarkable performance in computer vision and other different areas in the last recent years. Particularly, 3DCNNs—an extension of CNN—focus on extracting spatial and temporal features from videos. Some interesting applications that have been implemented using 3DCNN include object recognition [37], human action recognition [38], gesture recognition [39], and —particularly related to this work— behavior analysis from customers in the baking sector [40]. Although all the works mentioned before involve using a 3DCNN, each one has a particular architecture and corresponding set of parameters to adjust. For example, concerning the number of layers, many approaches rely on simple structures that consist of two or three layers [37,38,41], while others require several layers for exhaustive learning [42–45]. Recently, Alfaifi and Artoli [46] proposed combining 3D CNN and LSTM for human action prediction (HAP). In their approach, the 3D CNN was used for feature extraction while the LSTM for classification. The model's strength relies on the robustness to pose, illumination, and surrounding clutter. This robustness allows predicting human activity accurately. The architecture consists of a single 3D Conv layer, a 3D max-pooling layer, an long short-term memory (LSTM) layer, and two fully connected layers for classification.

Concerning shoplifting, the current literature is somewhat limited. Surveillance material is, in most cases, a company's private property. The latter restricts the amount of data available for training and testing new surveillance models. For this reason, several approaches focus on training to detect normal behavior [47–50]. Anything that lies outside the cluster is considered abnormal. In general, surveillance videos contain only a small fraction of crime occurrences. Then, most of the videos in the data are likely to contain normal behavior. Many approaches have experienced problems regarding the limited availability of samples and their unbalanced category distribution. For this reason, some works have focused on developing models that learn with a minimal amount of data. As a reference, we include some representative works on the area of behavior detection in Table 1.

Table 1. Some relevant examples of works related to behavior detection, their dataset size, and the number of criminal videos considered. * Ko and Sim's work [51] presents many incident videos due to their behaviors as abnormal; however, they do not represent a crime itself.

Paper	Behaviors to Detect	Dataset Size	Criminal/Incident Videos
Bouma et al. [52]	Theft and pickpocketing	8 videos	5
Ishikawa and Zin [31]	Loitering	6 videos	6
Koller et al. [53]	Theft	12 videos	12
Tsushita and Zin [22]	Snatch theft	19 videos	9
Grant and Williams [54]	Violent crimes against people or property	24 videos	12
Koller et al. [55]	Bomb and theft	26 videos	18
Ko and Sim [51]	Hand shaking, hugging, kicking, punching, pointing, and pushing	50 videos	50 *
Troscianko et al. [56]	Fights, assaults, car crimes and vandalism	100 videos	18

Our work aims at developing a support approach for shoplifting crime prevention. Our model detects a person that, according to their behavior, is likely to commit a shoplifting crime. We achieve the latter by analyzing the people's comportment in the videos before the crime occurs. To the best of our knowledge, this is the first work that analyzes behavior to anticipate a potential shoplifting crime.

3. Methodology

As part of this work, we propose a methodology to extract segments from videos where people exhibit behaviors relevant to shoplifting crime. The methodology considers both normal and suspicious behaviors, being the task to classify them accordingly. The following lines describe the dataset used and how we split it for experimental purposes, the pre-crime Behavior (PCB) method, and the 3DCNN architecture used for feature extraction and classification.

3.1. Description of The Dataset

Among the many works related to surveillance security, the analysis of non-verbal behavior is one of the less researched areas [57]. This generates a lack of enhancement of security protocols and available information. Many works build their datasets using actors. However, they cannot catch the essential behavioral cues that an offender may show in a stressful situation. Some types of crimes have been more explored, such as crowd behavior, vandalism, fights, or assaults. For non-violent crimes, such as shoplifting, pickpocketing, or theft, it is harder to detect the crime in public places and get access to the videos.

In this work, we use the UCF-Crime dataset [28] to analyze suspicious behavior during the build-up of a shoplifting crime. The dataset consists of 1900 real-world surveillance videos and provides around 129 h of videos. The videos have not been normalized in length and present a resolution of 320 $\times$ 240 pixels. The dataset includes scenarios from several people and locations, which are grouped into 13 classes such as "abuse", "burglary", and "explosion", among others. We extracted the samples used in this investigation from the "shoplifting" and "normal" classes from the UCF-Crime dataset.

To feed our model, we require videos that show one or more people whose activities are visible before the crime is committed. Due to these restrictions, not all the videos in

the dataset are useful. Suspicious behavior samples were extracted only from videos that exhibit a shoplifting crime, but to be used by our system, such samples must not contain the crime itself. Conversely, normal behavior samples were extracted from the "normal" class. Thus, it is important to stress that the model we propose is a behavior classifier (normal or suspicious) and not a crime classifier.

For processing the videos and extracting the suspicious behavior samples (video segments that exhibit suspicious behavior), we propose a novel method, the Pre-Crime Behavior (PCB) method, which we explain in the next section. Once we obtain the suspicious behavior samples, we applied some transformations to produce several smaller datasets. First, to reduce the computational resources required for training, all the frames were transformed into grayscale and resized to four resolutions: 160×120, 80×60, 40×30, and 32×24 pixels. As, by summing up normal and suspicious behavior samples, we get 120 samples, we applied a flipping procedure to increase such a number. Such a flipping procedure consists of turning over each frame of the video sample horizontally, resulting in a video where the actions happen in the opposite direction.

Data augmentation techniques aim to increase the number of useful examples in the training dataset, producing variations of the original images that the model is likely to see. Examples of these techniques include flipping, rotation, zoom, and brightness. It is relevant to mention that many of these techniques are not useful in our work. For example, vertical flipping an image makes no sense in our system as the videos will never be watched upside down. Rotation turns the image clockwise an arbitrary number of degrees, but it may drop pixels out of the image and produce areas with no pixels, which have to be filled in somehow. Zoom augmentation either adds new pixels around the image (zoom out) or leaves out part of the original image (zoom in), leading to losing or altering the scene's information. The situations derived from using such data augmentation techniques could potentially do more harm than good and, for that reason, were not considered for this work. Given the reasons mentioned above, we considered that sticking only to horizontal flips was the most suitable strategy for this work. It generates additional training samples without adding or subtracting any information to the samples.

3.2. The Pre-Crime Behavior Method

Video sample segmentation does not follow a specific methodology in criminal intentions and suspicious behavior analysis. This makes it unreliable for creating a benchmark and testing a model across different video sets. For example, in some investigations, the segmentation is left to the experts' judgement [53,55]. In others, the researchers select the frame before the criminal act [54,56]. In some particular cases, there is no segmentation at all [22,31,51].

The Pre-Crime Behavior (PCB) method arises as a new proposal to unify moments, such as the build-up phase and the crime itself, and provide a new segment to the analysis, the suspect's behavior before any aggression attempt. It is composed of four steps that allow the identification of four specific moments in the video sample. The PCB method is described as follows.

1. Identify the instant where the offender appears for the first time in the video. We refer to this moment as the First Appearance Moment (FAM). The analysis of suspicious behavior starts from this moment.
2. Detect the moment when the offender undoubtedly commits a crime. This moment is referred to as the Strict Crime Moment (SCM). This moment contains the necessary evidence to argue the crime commission.
3. Between the FAM and the SCM, find the moment where the offender starts acting suspiciously. The Comprehensive Crime Moment (CCM) starts as soon as we detect that the offender acts suspiciously in the video.
4. After the SCM, locate the moment where the crime ends (when everything seems to be ordinary again). If the video sample started from this instant, we would have no

evidence of any crime committed in the past. This moment is known as the Back to Normality Moment (B2NM).

Please note that, as a sample video from the UCF-Crime dataset may contain more than one crime, the PCB method is applied once for each crime occurrence. Then, sometimes we can extract various suspicious behavior samples from the same video in the UCF-Crime dataset.

The output of the PCB method comprises four moments per crime in the input video. These four moments divide each sample into three relevant segments, as described below.

Pre-Crime Behavior Segment (PCBS). The PCBS is the video segment between the FAM and the CCM. This segment has the information needed to study how people behave before committing a crime, even acting suspiciously. Most human observers will fail to predict that a crime is about to occur by only watching the PCBS.

Suspicious Behavior Segment (SBS). The SBS is the video segment contained between the CCM and the SCM. The SBS provides specific information about an offender's behavior before committing a crime.

Crime Evidence Segment (CES). The CES represents the video segment included between the SCM and the B2NM. This segment contains the evidence to accuse a person of committing a crime.

For the sake of clarity, we present the four moments and the three segments derived from the PCB method graphically, as depicted in Figure 2.

Figure 2. Video segmentation by using the moments obtained from the Pre-Crime Behavior Segment (PCB) method.

To extract the samples from the videos, we follow the process depicted in Figure 3. Given a video that contains one or more shoplifting crimes, we identify the precise moment when the offense is committed. After that, we label the different suspicious moments—moments where a human observer doubts what a person in the video is doing. Finally, we select the segment before the suspect is preparing to commit the crime. These segments become the training samples for the Deep Learning (DL) model.

In a video sample, each segment has particular importance regarding the information it contains (see Figure 2). The PCB segment has less information about the crime itself, but it allows us to analyze the suspect's normal-acting behavior when they appears for the first time, even far from a potential crime. The SBS allows us to have a more precise idea about who may commit the crime, but it is not conclusive. Finally, the CES contains the doubtless evidence about a person committing a shoplifting crime. If we remove both the SBS and the CES from the video, the result will be a video containing only people shopping, and there will be no suspicion or evidence that someone commits a crime. That is the importance of

the accurate segmentation of the video. From the end of a CES until the next SBS, there is new evidence about how a person behaves before attempting a shoplifting crime.

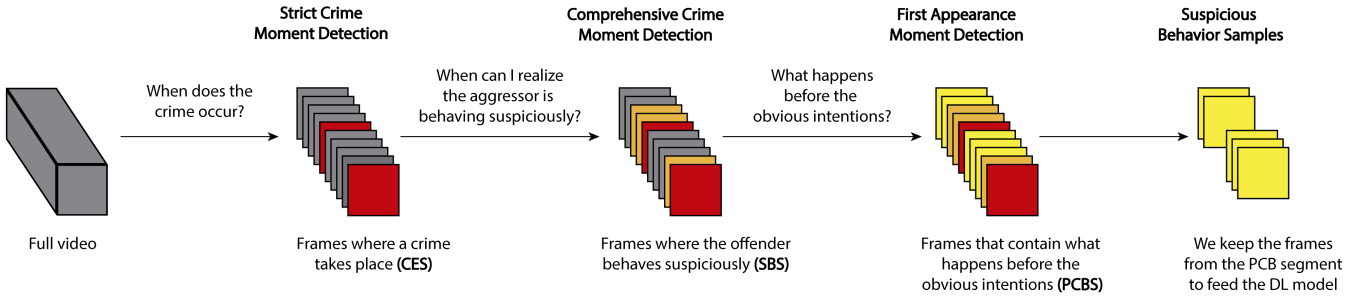


Figure 3. Graphical representation of the process for suspicious behavior sample extraction.

For experimental purposes, we only use the frames from the PCBS in this work. As these segments lack specific criminal behavior, they have no information about any transgression. The PCB segments are ideal for feeding our 3DCNN model, aiming to characterize the people's behavior. The objective of the model is to identify when such behavior is suspicious, which may indicate that a shoplifting crime is about to be committed.

3.3. 3D Convolutional Neural Networks

For this work, we use a 3DCNN for feature extraction and classification. 3DCNN is a recent approach for spatio-temporal analysis that has shown remarkable performance in processing videos in different areas, such as moving objects action recognition [37], gesture recognition [39], and action recognition [38]. We decided to implement a 3DCNN in a more challenging context, such as searching for patterns in video samples, which lack suspicious and illegal visual behavior.

We employ a basic structure to explore the performance of the 3DCNN for behavior classification. The model comprises four Conv3D layers (two pairs of consecutive convolutional layers for capturing long dependencies [43,58,59]), two max-pooling layers, and two fully connected layers. As a default configuration, in the first pair of Conv3D layers, we apply 32 filters, and for the second pair, 64 filters. All kernels have a size of $3 \times 3 \times 3$, and the model uses an Adam optimizer and cross-entropy for loss calculation. The graphical representation of this model is shown in Figure 4. The last part of the model contains two dense layers with 512 and two neurons, respectively. This architecture was selected because it has been used for similar applications [60], and it seems suitable as a first approach for behavior detection in surveillance videos.

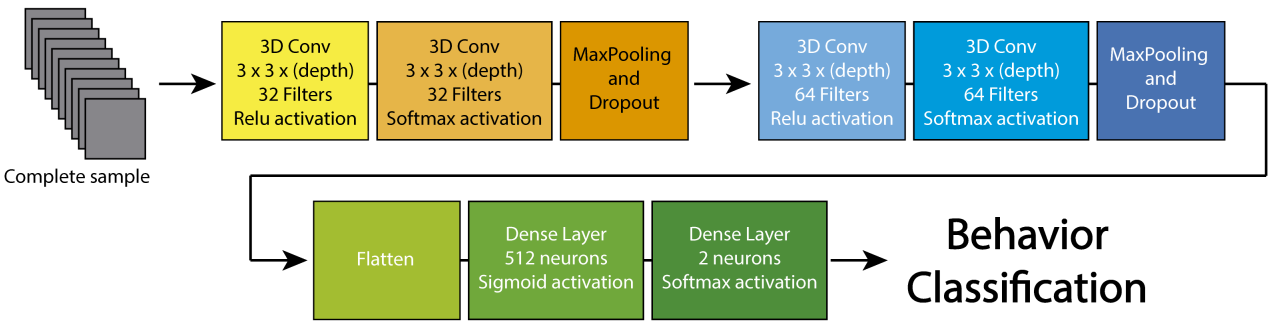


Figure 4. Architecture of the DL Model used for this investigation. The depth of the kernel for the 3D convolution is adjusted to 10, 30, or 90 frames, according to each particular experiment (see Section 4).

For handling the model training, we use Google Colaboratory [61]. This free cloud tool allows us to write and execute code in cells and runs directly on a browser to train DL models. We can upload the datasets to a storage service, link the files, prepare the training environment, and save considerable time during the model training using a virtual GPU.

3.4. Metrics

As the decisive metric to analyze the results, we considered the accuracy (Equation (1)). It considers the correct hits—true positive (TP) plus true negative (TN)—over the total number of samples evaluated (FP and FN represent false positives and false negatives, respectively).

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

As accuracy shows the general performance of the model, we complement its information by presenting the confusion matrices of the best runs. These matrices allow checking, in detail, the model capability to classify suspicious and normal behavior. We used two additional metrics for adequately analyzing the results from the confusion matrices: precision (Equation (2)) and recall (Equation (3)). Precision indicates the proportion of samples classified as suspicious that are, in fact, suspicious—a model with a precision of 1.0 produces no FP. Recall indicates the proportion of actual suspicious samples that were correctly classified by the system—a model with a recall of 1.0 produces no FN.

$$precision = \frac{TP}{TP + FP} \tag{2}$$

$$recall = \frac{TP}{TP + FN} \tag{3}$$

4. Experiments And Results

We conducted a total of six experiments in this work. These experiments are divided into two categories: preliminary and confirmatory. The first four experiments are preliminary as they focus on exploring the effect of different configurations under different scenarios, aiming to find some suitable configurations that may lead to better model performance. We refer to the last two experiments as confirmatory as we tested the system on more challenging configurations derived from the preliminary experiments, to validate the approach. Among all these experiments, a total of 708 models were generated and tested. Although the specific details of each experiment are detailed in its corresponding description, for the ease of the reader, we have provided an overview of our experimental setup in Figure 5.

4.1. Preliminary Experiments

In this set of experiments, we explore different configurations for the system and estimate their effect on its overall performance in terms of the accuracy obtained. The rationale behind this first set of experiments is that we affect the model's performance by introducing small variations on its parameters. Then, finding a good set of input parameters is a way to improve the overall performance of the model.

For this work, four parameters have been considered for tuning purposes. These parameters, as well as their available values, are listed below.

Training set size. The percentage of the samples from the base dataset used for training. The possible values are 80%, 70%, and 60%. Note that, as an attempt to test out approach on different situations, the base dataset changes for each particular experiment.

Depth. The number of consecutive frames used for 3D convolution. The values allowed for this parameter are 10, 30, and 90 frames.

Resolution. The size of the input images (in pixels). We used four different values for resolution: 32 × 24, 40 × 30, 80 × 60, and 160 × 120 pixels.

Flip. As discussed before, to increase the number of samples, we applied a horizontal flipping procedure to all the samples. By using this procedure, we doubled the number of samples. Stating that a set has been flipped indicates that the frames in those videos have been flipped horizontally.

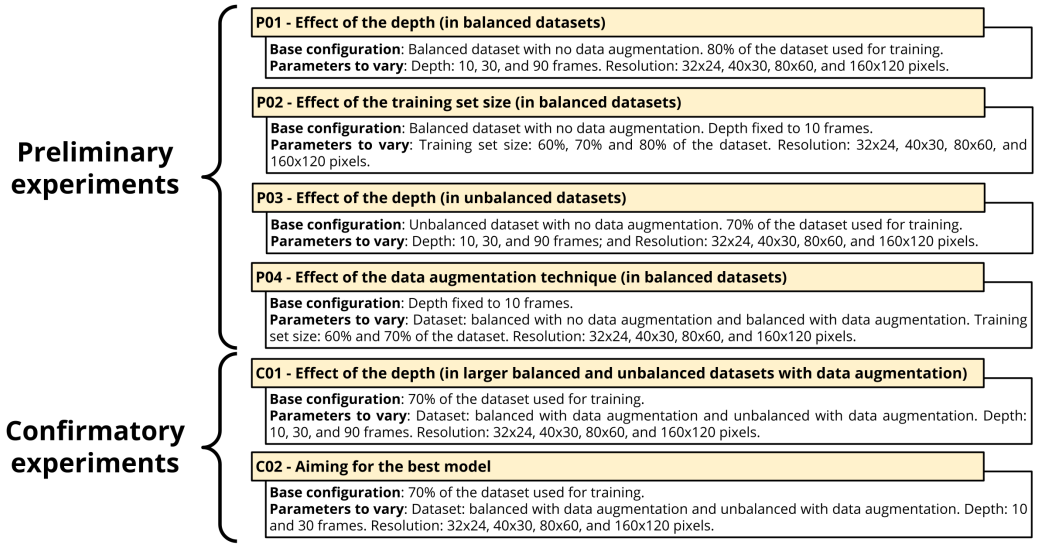


Figure 5. Overview of the experimental setup followed in this work. For a detailed description of the parameters and the relation of the samples considered for each experiment, please consult Appendix A.

To evaluate the impact of varying the values for these parameters, we conducted five independent experiments. Each of these experiments followed a factorial design with two factors (one factor per parameter). In all the experiments, one of the factors was always the resolution in pixels. We trained three independent models per combination of such factors. The results are analyzed both from the statistical perspective (main effects and interaction effects through a two-way ANOVA) and the practical one (by analyzing the interaction plots and the average accuracy derived from the observations). It is relevant to mention that all the cases satisfied both normality and homogeneity of the variances, which are conditions required to apply the two-way ANOVA. We tested normality by analyzing the residuals and through the Shapiro–Wilk test of normality, while we applied the Levene's test to check the homogeneity of variances within groups. The significance value considered for all the statistical tests in this work was 5%.

4.1.1. Experiment P01—Effect of the Depth (In Balanced Datasets)

To analyze the impact of varying the depth size (number of consecutive frames to consider) under different resolutions, we analyzed its effect on a balanced dataset containing 30 normal behavior samples and 30 suspicious behavior ones, where 80% of those samples were used for training the models. As there were, as explained before, three values allowed for the depth: 10, 30, and 90 frames, and four values for the resolution: 32 × 24, 40 × 30, 80 × 60, and 160 × 120 pixels, the combinations of these parameters resulted in 12 configurations. For each configuration, we conducted three independent runs to generate three models. With this data, we ran a two-way ANOVA to analyze the effect of depth and resolution on the accuracy of the model, given the base configuration. Table 2 presents the accuracy of the three independent models trained for each configuration and their average accuracy per configuration.

Table 2. Accuracy (%) of the models trained with the 12 different configurations from experiment P01. Each cell presents the accuracy of three independent models per configuration and its average. The best results per resolution are highlighted in bold.

	Resolution (Pixels)			
Depth (Frames)	**32 × 24**	**40 × 30**	**80 × 60**	**160 × 120**
10	83.3	75.0	66.6	50.0
	75.0	66.6	83.3	50.0
	91.6	75.0	83.3	41.6
	83.3	72.2	**77.7**	69.3
30	83.3	83.3	83.3	83.3
	75.0	66.6	75.0	50.0
	50.0	75.0	50.0	75.0
	69.4	**75.0**	69.4	**69.4**
90	83.3	66.6	50.0	50.0
	75.0	75.0	75.0	50.0
	50.0	50.0	58.3	75.0
	69.4	63.9	61.1	58.3

By analyzing the statistical results, we found that neither the main effects are significant, nor their interaction (with a significance level of 5%). The *p*-values obtained from the two-way ANOVA for the main effects of depth and resolution in this experiment were 0.1044 and 0.6488, respectively. The *p*-value for their interaction was 0.9502. As there is no statistical evidence that suggests that changes in the depth or the resolution affect the accuracy of the model, we extended the analysis and considered inspecting the interaction plot, which is depicted in Figure 6. This interaction plot suggests that independently of the depth considered, using 160 × 120 pixels as resolution obtains the worst results. Furthermore, based only on the 36 observations analyzed, using ten frames and 32 × 24 pixels obtains the best average results on the fixed values used for this experiment. For this reason—and based on the fact that we could not derive any other conclusion from the statistical perspective—we considered using ten frames as the best value for depth for the next set of experiments.

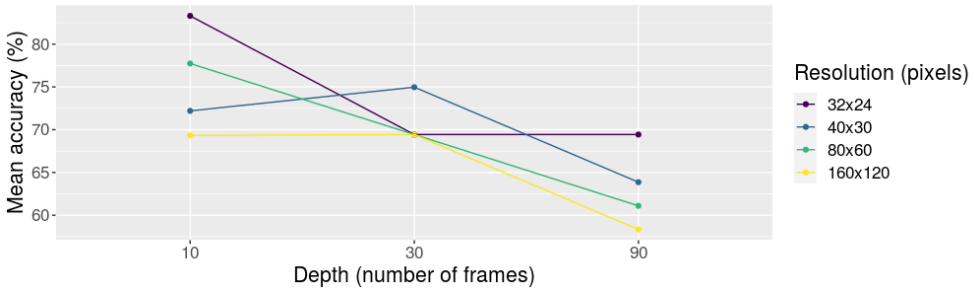


Figure 6. Interaction plot of depth (10, 30, and 90 frames) and resolution (32 × 24, 40 × 30, 80 × 60, and 160 × 120 pixels) using the accuracy values obtained from experiment P01.

4.1.2. Experiment P02—Effect of the Training Set Size (In Balanced Datasets)

In this experiment, we changed the proportion of samples used for training, combined with four values of the resolution, as an attempt to estimate their effect on the model's accuracy. As in the previous experiment, the dataset is also balanced with 30 normal behavior samples and 30 suspicious behavior ones. However, for this experiment, the depth parameter was fixed to 10. This value was taken from the previous experiment since we obtained the best results by using ten frames. We allowed three values for defining the

training set size: 80%, 70%, and 60% of the total of samples in the dataset (that contains 60 samples as previously described). As in the previous experiment, the available values for the resolution were 32 × 24, 40 × 30, 80 × 60, and 160 × 120 pixels. For each configuration, we trained three independent models and used their accuracy to run a two-way ANOVA to analyze both the main and interaction effects of the two variables, given the base sample set. The accuracy of the three independent models per configuration is shown in Table 3.

Table 3. Accuracy (%) of the models trained with the 12 different configurations from experiment P02. Each cell presents the accuracy of three independent models per configuration and its average. The best results per resolution are highlighted in bold.

	Resolution (Pixels)			
Training	**32 × 24**	**40 × 30**	**80 × 60**	**160 × 120**
80%	66.6	75.0	66.6	50.0
	75.0	75.0	58.3	50.0
	75.0	66.6	75.0	41.6
	72.2	72.2	66.6	47.2
70%	77.7	72.2	66.6	77.7
	66.6	77.7	72.2	77.7
	61.1	72.2	66.6	72.2
	68.5	**74.0**	**68.5**	**75.9**
60%	62.5	66.6	70.8	72.2
	58.3	66.6	50.0	66.6
	70.8	70.8	62.5	72.2
	63.9	68.0	61.1	70.3

The statistical analysis through a two-way ANOVA showed that the main effects, the proportion of samples used for training and the resolution, are not significant with $\alpha = 0.05$ (the p-values were 0.0140 and 0.0771, respectively). However, their interaction was statistically significant, with a p-value of 0.0004. Because the interaction effect was statistically significant, we compared all group means from the interaction of the two factors. The p-values were adjusted by using the Tukey method for comparing a family of 12 configurations. The results suggested that the worst combination arose when using 80% of the base dataset for training and 160 × 120 pixels as resolution. The confidence interval for the average accuracy (with 95% of confidence), lies between 36.6% and 57.8%. Conversely, the remaining configurations are considered equally useful from the statistical perspective, since their confidence intervals overlap.

To have a better look at the behavior of these configurations, we also analyzed the interaction plot of the proportion of samples from the base dataset used for training and the resolution (Figure 7). The interaction plot confirmed the idea that using 80% and 160 × 120 pixels harms the process. So far, we do not have an explanation for such behavior yet. However, we can also observe how similar the remaining configurations are, in terms of accuracy. As the best results in this experiment were obtained by using 70% of the base dataset for training purposes, we kept this value as a recommended one for the following experiments.

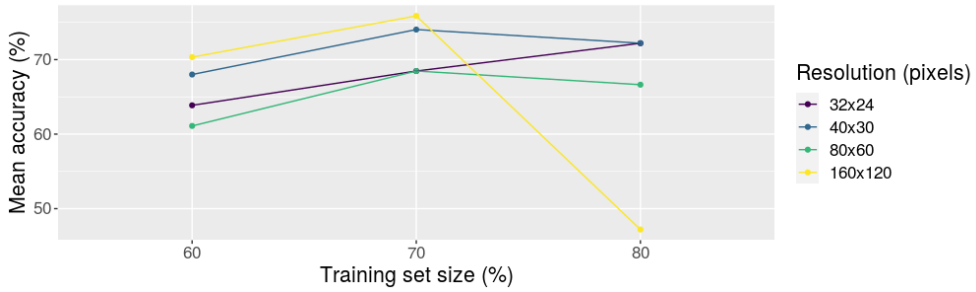


Figure 7. Interaction plot of the proportion of the base set used for training (80%, 70%, and 60%) and resolution (32 × 24, 40 × 30, 80 × 60, and 160 × 120 pixels) using the accuracy values obtained from experiment P02.

4.1.3. Experiment P03—Effect of the Depth (In Unbalanced Datasets)

At this point, we had only explored the behavior of the models in balanced sets (same proportion of normal behavior and suspicious behavior samples). For this experiment, we analyzed the effect of the depth and the resolution (as we did in experiment P01), but this time on an unbalanced set that contains 90 samples (60 normal behavior samples and 30 suspicious behavior ones). As we learned from the previous experiment, the models obtained the best performance when 70% of the base dataset was used for training. Then, we used such a value for this experiment. For the depth, three values were allowed: 10, 30, and 90 frames, while four values were available for the resolution: 32 × 24, 40 × 30, 80 × 60, and 160 × 120 pixels. The combinations of these parameters give 12 configurations. For each of these configurations, we trained three independent models. The results from this experiment, in terms of accuracy, are depicted in Table 4.

The statistical analysis through the two-way ANOVA suggests that the effect of the depth is not statistically significant (p-value of 0.1786). However, the effect of the resolution, as well as the interaction between the depth and the resolution, are statistically significant with p-values of 7.09×10^{-5} and 0.0031, respectively. As the interaction between the depth and the resolution is important in this case, we used the Tukey method for comparing a set of 12 configurations and adjusting the p-values, as we did in the previous experiment. The results show that the configurations can be classified into four groups, based on the accuracy obtained. However, these groups overlap for many of the configurations. Based on the confidence intervals for the average accuracy of the models (with 95% of confidence), the configuration with the most promising confidence interval for the average accuracy was using 90 frames and 80 × 60 pixels as resolution.

For clarity, we also included the interaction plot as we did for the previous experiments. Figure 8 suggests that, in unbalanced sets, the combination of depth and resolution is important to get a good accuracy. The information from the interaction plot seems to indicate that, for large resolutions such as 160 × 120 pixels, increasing the number of frames decreases the model's accuracy. Conversely, for slightly lower resolutions such as 80 × 60 pixels, increasing the number of frames improves the model's accuracy. Then, based on the statistical results as well as the analysis of the interaction plot, we can recommend that, when dealing with unbalanced sets, the best configuration is to use 90 frames and 80 × 60 pixels.

Table 4. Accuracy (%) of the models trained with the 12 different configurations from experiment P03. Each cell presents the accuracy of three independent models per configuration and its average. The best results per resolution are highlighted in bold.

	Resolution (Pixels)			
Depth (Frames)	**32 × 24**	**40 × 30**	**80 × 60**	**160 × 120**
10	66.6	70.3	62.0	74.1
	66.6	62.9	66.6	77.7
	66.6	70.3	77.7	85.1
	66.6	**67.8**	68.8	79.0
30	70.3	55.5	81.4	77.7
	66.6	66.6	77.7	77.7
	70.3	74.0	81.4	85.1
	69.1	65.4	80.2	**80.2**
90	66.6	62.9	81.4	66.6
	70.3	62.9	81.4	66.6
	70.3	70.3	81.4	66.6
	69.1	65.4	**81.4**	66.6

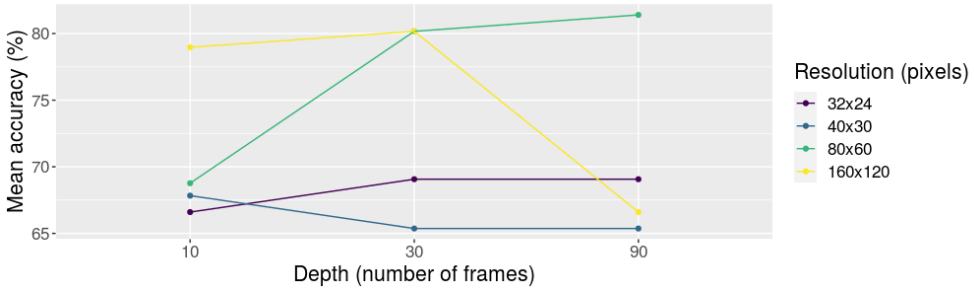


Figure 8. Interaction plot of depth (10, 30, and 90 frames) and resolution (32 × 24, 40 × 30, 80 × 60, and 160 × 120 pixels) using the accuracy values obtained from experiment P03.

4.1.4. Experiment P04—Effect of the Data Augmentation Technique (In Balanced Datasets)

Data augmentation techniques are an option to take advantage of small datasets. For this reason, we tested the model performance using original and horizontally flipped images in different runs. The training set has a size of 60% (Table 5 and Figure 9) and 70% (Table 6 and Figure 10) of the total dataset.

When 60% of the dataset was used for training, we found that the effect of using data augmentation is not significant (its p-value was 0.1736). However, the effect of the resolution is significant, given a p-value of 0.0050. The p-value for the interaction of these two factors was 0.0176, which is significant, with a 5% of significance. To extend the analysis, we also provide the interaction plot of these two factors, which is depicted in Figure 9. As it can be observed, including the horizontally flipped samples, in general, increases the model's performance. The only configuration that seems to contradict this trend is when the resolution is set to 160 × 120 pixels. We do not have a concrete explanation of this behavior, but it could be related to the computing power related to learning at a higher resolution.

Table 5. Accuracy (%) of the models trained with the eight different configurations from experiment P04 (using 60% of the dataset for training). Each cell presents the accuracy of three independent models per configuration and its average. The best results per resolution are highlighted in bold.

	Resolution (Pixels)			
Flipped	**32 × 24**	**40 × 30**	**80 × 60**	**160 × 120**
FALSE	72.9	70.8	79.1	77.0
	70.8	72.9	79.1	83.3
	70.8	70.8	72.9	70.8
	71.5	71.5	77.0	**77.0**
TRUE	70.8	77.0	83.3	72.9
	75.0	75.0	87.5	68.7
	75.0	79.1	79.1	70.8
	73.6	**77.0**	**83.3**	70.8

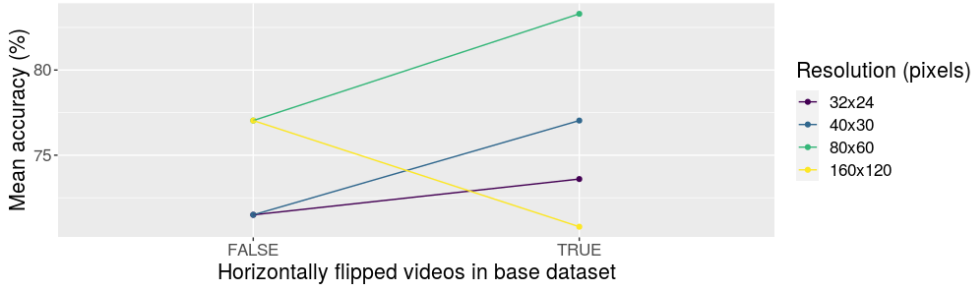


Figure 9. Interaction plot of depth (10, 30, and 90 frames) and resolution (32 × 24, 40 × 30, 80 × 60, and 160 × 120 pixels) using the accuracy values obtained from experiment P04 (using 60% of the dataset for training).

When 70% of the dataset was used for training, we could not found evidence that the main effects, and neither their interaction, were significant. However, the effect of the resolution is significant with $\alpha = 0.05$ (the p-values were 0.0.1797, 0.1051, and 0.2248, respectively). To deepen this situation, we present the interaction plot of these two factors, which is depicted in Figure 10. Based on the interaction plot, using 70% of the dataset when horizontally flipped samples are included does not affect the model's performance. The only configuration that seems to contradict this idea is when the resolution is set to 80 × 60 pixels. In this case, we cannot explain why this situation occurred. Further research in this regard will be needed to explain this behavior.

Table 6. Accuracy (%) of the models trained with the eight different configurations from experiment P04 (using 70% of the dataset for training). Each cell presents the accuracy of the three independent models per configuration and its average. The best results per resolution are highlighted in bold.

	Resolution (Pixels)			
Flipped	**32 × 24**	**40 × 30**	**80 × 60**	**160 × 120**
FALSE	69.4	66.6	72.2	72.2
	77.7	72.2	75	83.3
	80.5	75.0	66.6	83.3
	75.9	71.3	71.3	**79.6**
TRUE	80.5	66.6	75.0	77.7
	75	77.7	86.1	77.7
	75	72.2	83.3	80.5
	76.8	**72.2**	**81.5**	78.6

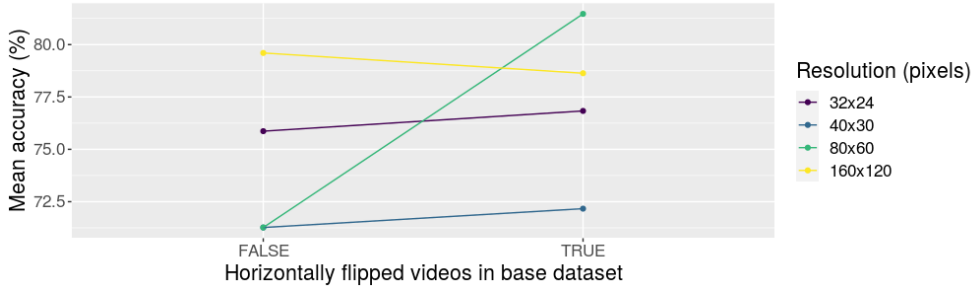


Figure 10. Interaction plot of depth (10, 30, and 90 frames) and resolution (32 × 24, 40 × 30, 80 × 60, and 160 × 120 pixels) using the accuracy values obtained from experiment P04 (using 70% of the dataset for training).

As the experiments with 60% and 70% obtained similar results in terms of accuracy, we considered that any of these configurations could effectively be used to train the model. Then, we kept 70% as the proportion of samples to be used for training in the remaining experiments.

4.2. Confirmatory Experiments

For the confirmatory experiments, we focused on analyzing our model on larger datasets that include horizontally flipped samples. For this purpose, we analyzed the effect of the depth and resolution (as we did in experiment P01) but this time on larger sets that included horizontally flipped samples. The first dataset contains 240 samples (120 normal behavior samples and 120 suspicious behavior ones) while the second contains 180 samples (120 normal behavior samples and only 60 suspicious behavior ones). However, these samples are not independent as they include flipped ones. When we refer to the number of samples, we mean the number of available videos, regardless of being the original ones extracted using the PCB method or their flipped versions. Based on the proportion of normal and suspicious samples in each dataset, we can state that the first one is "balanced", while the second one is not. As we learned from the previous experiment, the training set corresponds to 70% of the base dataset.

4.2.1. Experiment C01—Effect of the Depth (In Larger Balanced and Unbalanced Datasets with Data Augmentation)

First, for the depth, three values were allowed: 10, 30, and 90 frames, while four values were available for the resolution: 32 × 24, 40 × 30, 80 × 60, and 160 × 120 pixels. Then, for each set, the combinations of depth and resolution produce 12 different configurations. The results of this experiment are depicted in Tables 7 and 8.

In the case of the balanced dataset (Table 7), the best average results are mainly obtained when 30 frames are used. If we consider the resolution, it seems that 80 × 60 is the best choice. In general, it seems that using 90 frames affects the model's performance. The results are similar when the unbalanced dataset is used (Table 8). However, this time there is one case where using 90 frames produced the best average results (when combined with 40 × 30 pixels as resolution).

Table 7. Accuracy (%) of the models trained with the 12 different configurations from experiment C01 (balanced dataset with data augmentation). Each cell presents the accuracy of the three independent models per configuration and its average. The best results per resolution are highlighted in bold.

	Resolution (Pixels)			
Depth (Frames)	**32 × 24**	**40 × 30**	**80 × 60**	**160 × 120**
10	75.0	72.2	83.3	77.7
	84.7	86.1	86.1	77.7
	66.6	68.0	91.6	80.5
	75.4	75.4	**87.0**	78.6
30	80.5	66.6	76.3	86.1
	77.7	80.5	86.1	90.2
	75.0	81.9	75.0	81.9
	77.7	**76.3**	79.1	**86.1**
90	69.4	72.2	83.3	50.0
	75.0	79.1	81.9	77.7
	79.1	75.0	83.3	50.0
	74.5	75.4	82.8	59.2

Table 8. Accuracy (%) of the models trained with the 12 different configurations from experiment C01 (unbalanced dataset with data augmentation). Each cell presents the accuracy of the three independent models per configuration and its average. The best results per resolution are highlighted in bold.

	Resolution (Pixels)			
Depth (Frames)	**32 × 24**	**40 × 30**	**80 × 60**	**160 × 120**
10	83.3	68.5	81.4	79.6
	64.8	70.3	77.7	77.7
	57.4	66.6	79.6	79.6
	68.5	68.5	**79.6**	**79.0**
30	72.2	66.6	87.0	87.0
	81.4	70.3	61.1	74.0
	74.0	62.9	81.4	68.5
	75.9	66.6	76.5	76.5
90	68.5	74.0	59.2	70.3
	83.3	74.0	70.3	66.6
	70.3	72.2	81.4	66.6
	74.0	**73.4**	70.3	67.8

4.2.2. Experiment C02—Aiming for the Best Model

Based on the results from the previous experiment, we analyzed the results to decide which parameters might improve behavior classification, and selected the configurations with the best performance. Then, the depth of 90 frames was excluded from this experiment. We repeated the configurations used in the previous experiment (excluding the 90 frames as depth), but this time running 30 times each configuration. Besides, this time we used using cross-validation, to extend the results previously obtained.

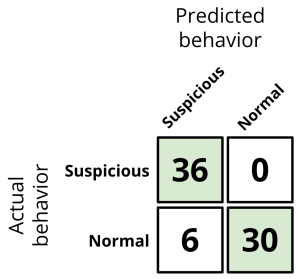
Table 9 presents average accuracy and the standard deviation of each configuration tested in experiment C02. Most of the results have an accuracy of around 70%. As observed, there is no significant deviation in each training group. The results seem very similar among them. However, the results when 10 frames and 80 × 60 pixels are used is slightly

better than the rest. On an individual level, the best model was also obtained when this resolution was used. The best model correctly classified 92.50% of the samples.

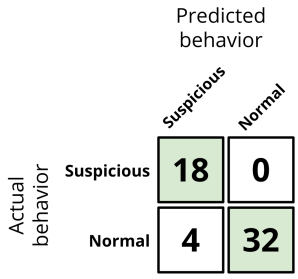
On a final test, we used the best model obtained to solve the four configurations available when 80 × 60 pixels are used. This way, we tested the model on the balanced dataset with 10 and 30 frames, and in the unbalanced dataset, also with 10 and 30 frames. The results are presented in terms of the confusion matrices, as shown in Figure 11. To deepen the results, we present the precision and recall for each class in isolation. For suspicious behavior, the model presents a precision that ranges from 0.7826 (unbalanced dataset with 30 frames) to 0.8571 (balanced dataset with 10 frames). This means that when the model classifies a behavior as suspicious, it is correct in at least 78% of the cases. Regarding the recall for suspicious behavior, it is equal to 1 in all cases. This means that the model correctly classifies all the suspicious behavior samples in the test set (no suspicious sample was classified as a normal one). For normal behavior, the model's precision is always equal to 1, meaning that whenever the model predicts that a sample is normal, the model is always correct. Regarding the recall for normal behavior, the values range from 0.8055 (balanced dataset with 30 frames) to 0.8888 (unbalanced dataset with 10 frames), which means that the best model correctly classifies 88% of the normal behavior samples in the unbalanced dataset with 10 frames of depth.

Table 9. Accuracy (%) of the models trained with the 16 different configurations from the confirmatory experiment. Each cell presents the average accuracy of 30 independent models per configuration and its standard deviation. The best results per resolution are highlighted in bold.

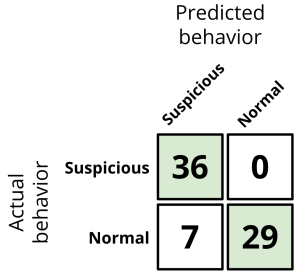
		Resolution (Pixels)			
Number of Samples (Normal/Suspicious)	**Depth (Frames)**	**32 × 24**	**40 × 30**	**80 × 60**	**160 × 120**
120/120	10	70.3 (0.0476)	71.8 (0.0468)	73.0 (0.0717)	73.1 (0.0661)
	30	70.1 (0.0574)	**71.9 (0.055)**	73.6 (0.0821)	71.6 (0.0999)
120/60	10	69.4 (0.0686)	68.7 (0.0569)	**75.0 (0.0689)**	**75.7 (0.0638)**
	30	**71.6 (0.0533)**	69.1 (0.0576)	74.8 (0.0500)	73.9 (0.0543)



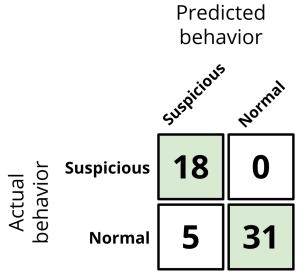
(**a**) Balanced dataset with 10 frames.



(**b**) Unbalanced dataset with 10 frames.



(**c**) Balanced dataset with 30 frames.



(**d**) Unbalanced dataset with 30 frames.

Figure 11. Confusion matrices for the best model generated for each configuration in the confirmatory experiment.

4.3. Discussion

There are some aspects related to the proposed model and the results obtained so far that are worth discussing:

- The system can be used to classify normal and suspicious behavior given the proper conditions.
- The PCB method exhibits some limitations as it is yet a manual process.
- The time needed for training the models suggests that training time may not be related to accuracy.
- There is an apparent relationship between the model's performance and the number of parameters in the models.

The following lines deepen into these critical aspects.

As the first experiment in this work, we selected a 3D Convolutional Neural Network with a basic configuration as a base model. Then, we tried different configurations as a means for parameter tuning. The result of this process was a configuration that improved the performance of the model. From the parameter exploration phase, we found that 80×60 and 160×120 resolutions delivered better results than a commonly used low resolution. This experiment was limited to a maximum resolution of 160×120 due to processing resources. Another significant aspect to consider is the "depth" parameter. This parameter describes the number of consecutive frames used to perform the 3D convolution. After testing different values, we observed that small values, between 10 and 30 frames, show a good trade-off between image detail and processing time. These two factors impact the network model training and the correct classification of the samples. Furthermore, the proposed model can correctly handle flipped images and unbalanced datasets. We confirmed this idea through the experiments performed on a more realistic simulation where the dataset has more normal behavior samples than suspicious behavior ones.

It is important to clarify the process for extracting the behavior samples from the UCF-Crime dataset. We are aware of the problems that may arise from using a non-automated method to extract the video segments from the original dataset. For example, (1) as it is a manual process, it is restricted to small datasets, and (2) due to its subjectivity, different executions may lead to different video segments (even if the same observer is involved). Although the PCB method exhibits those limitations, no other investigation has addressed this problem in the way we propose. Then, the PCB method is the only systematic technique we have to extract behavioral information in the way we need it, from the original dataset. For the sake of reproducibility, we have included a relation of the segments of videos from the UCF-Crime dataset that we used as input for the DL models in this work. This information can be consulted in the appendices, in Tables A1 and A2. Then, any future work that wants to use our video samples can use such segments—without the need to rerun the PCB method.

Regarding the processing time, we use Google Colaboratory to perform the experiments in this work. This tool is based on Jupyter Notebooks and allows using the GPU. The speed of each training depends on the tool demand. Most of the networks in this investigation were trained in less than an hour. However, a higher GPU demand may impact the training time. At the moment, we cannot establish a formal relationship between the resolutions of the videos and the training time, but we have an estimation of how different depths impacted the training time. Table 10 shows the average training times of models generated for experiment C01, as described in Section 4.2.1 (the results of these experiments are shown in Tables 7 and 8). From these results, it is clear that using a higher resolution and a larger depth increases the computational resources required for the training. In our particular case, some of the runs on the higher resolution (160×120) and maximum number of frames (90) took up to four hours. This information should be taken into consideration for further studies as the training time is an essential factor and, in this case, we are dealing with datasets that can be considered small. Besides, another point to consider is the model's accuracy against the training time required to generate such a model. Although the training time drastically increases when the resolution increases,

the accuracy does not increase in a similar proportion. Particularly, we found cases where increasing the resolution worsen the accuracy of the models produced.

As the results from the confirmatory experiments suggest, the 80×60 input resolution generates the best accuracy values. Although we have not confirmed our ideas, we think the accuracy might be related to the network's number of parameters and image information. A balance between these two parameters may impact the final result. While smaller resolutions mean fewer parameters, it also means less information to model the offender's behavior. On the contrary, a big resolution could give more details in visual data to analyze, but also imply more processing and many more parameters to optimize. Thus, we think this balance between resolution and parameters might cause an improvement in the model's accuracy. However, more research is required to support this claim.

Table 10. Average training times in seconds comparison between different depths and resolutions.

		Resolution (Pixels)			
Number of Samples (Normal/Suspicious)	**Depth (Frames)**	**32 × 24**	**40 × 30**	**80 × 60**	**160 × 120**
120/120	10	118	157	475	1714
	30	257	364	1304	4952
	90	688	1011	3879	15,415
120/60	10	96	126	369	1356
	30	196	279	1027	3918
	90	518	758	2929	11,655

5. Conclusions

For this work, we have focused on the behavior performed by a person during the build-up phase of a shoplifting crime. The neural network model identifies the previous conduct, looking for suspicious behavior, and not recognizing the crime itself. This behavior analysis is the principal reason why we remove the committed crime segment from the video samples, to allow the artificial model to focus on decisive conduct and not in the offense. We implement a 3D Convolutional Neural Network due to its capability to obtain abstract features from signals and images, based on previous action recognition and movement detection approaches.

Based on the results obtained from the conducted experimentation, 75% of accuracy in suspicious behavior detection. Then, we can state that it is possible to model a person's suspicious behavior in the shoplifting context. We found which parameters fit better for behavior analysis through the presented experimentation, particularly for the shoplifting context. We explore different parameters and configurations, and, in the end, we compare our results against a reference 3D Convolutional architecture. The proposed model demonstrates a better performance with balanced and unbalanced datasets using the particular configuration obtained from previous experiments.

The final intention of this experimentation is to develop a tool capable of supporting the surveillance staff, presenting visual behavioral cues, and this work is a first step to achieve the mentioned goal. We will explore different aspects that will contribute to the project development, such as bigger datasets, adding more criminal contexts that present suspicious behavior, and real-time tests.

In these experiments, we used a selected number of videos from the UCF-Crimes dataset. As future work, and aiming at testing our model in a more realistic simulation, we will increase the number of samples, preferably the normal behavior ones, to create a bigger sample imbalance between classes. Another exciting aspect of the development of this project is expanding our behavior detection model to other contexts. It exists many situations where we can find suspicious behavior, such as stealing, arson intents, and burglary.

We will gather videos of different contexts to strengthen the capability to detect suspicious behavior. Finally, the automation of the PCB method for video segmentation stands out as an interesting point to explore. This will reduce the preprocessing time, which would allow analyzing a larger amount of data. For this reason, we consider this an important path for future work derived from this investigation.

Author Contributions: Conceptualization, G.A.M.-M. and J.R.A.-P.; Data curation, G.A.M.-M.; Formal analysis, G.A.M.-M. and J.C.O.-B.; Investigation, G.A.M.-M.; Methodology, G.A.M.-M. and J.C.O.-B.; Software, G.A.M.-M., J.R.A.-P. and J.C.O.-B.; Supervision, J.C.O.-B., A.G.-C. and H.T.-M.; Writing—original draft, G.A.M.-M.; Writing—review and editing, J.R.A.-P., J.C.O.-B., A.G.-C. and H.T.-M. All authors have read and agreed to the published version of the manuscript.

Funding: The APC was funded by Tecnologico de Monterrey.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

3DCNN	Three-Dimensional Convolutional Neural Network
ACC	Accuracy
ANOVA	Analysis of Variance
B2NM	Back to Normal Moment
CCM	Comprehensive Crime Moment
CES	Crime Evidence Segment
DL	Deep Learning
FAM	First Appearance Moment
GPU	Graphics Processing Unit
NRSS	National Retail Security Survey
PCB	Pre-Crime Behavior method
PCBS	Pre-Crime Behavior Segment
SBS	Suspicious Behavior Segment
SCM	Strict Crime Moment
TN	True Negative
TP	True Positive
VFOA	Visual Focus Of Attention

Appendix A. Experimental Description

- **Preliminary experiment P01**:
  - Dataset: IDs 1 to 30 from Table A1 and IDs 1 to 30 from Table A2.
  - Training set size: 80% of the base dataset.
  - Depth: 10, 30, and 90 frames.
  - Resolutions: 32×24, 40×30, 80×60, and 160×120.
  - Epochs: 100.
  - Runs: Three per configuration.
- **Preliminary experiment P02**:
  - Dataset: IDs 1 to 30 from Table A1 and IDs 1 to 30 from Table A2.
  - Training set size: 80%, 70%, and 60% of the base dataset.
  - Depth: 10 frames.
  - Resolutions: 32×24, 40×30, 80×60, and 160×120.
  - Epochs: 100.
  - Runs: Three per configuration.
- **Preliminary experiment P03**:
  - Dataset: IDs 1 to 60 from Table A1 and IDs 1 to 30 from Table A2.
  - Training set size: 70% of the base dataset.
  - Depth: 10, 30, and 90 frames.

  - Resolutions: 32 × 24, 40 × 30, 80 × 60, and 160 × 120.
  - Epochs: 100.
  - Runs: Three per configuration.
- **Preliminary experiment P04**:
  - Datasets: IDs 1 to 60 from Table A1 and IDs 1 to 60 from Table A2; and IDs 1 to 60 from Table A1 (horizontally flipped) and IDs 1 to 60 from Table A2 (horizontally flipped).
  - Training set size: 60% and 70% of the base dataset.
  - Depth: 10.
  - Resolutions: 32 × 24, 40 × 30, 80 × 60, and 160 × 120.
  - Epochs: 100.
  - Runs: Three per configuration.
- **Confirmatory experiment C01**:
  - Datasets: IDs 1 to 60 from Table A1 and IDs 1 to 60 from Table A2, IDs 1 to 60 from Table A1 (horizontally flipped) and IDs 1 to 60 from Table A2 (horizontally flipped); IDs 1 to 60 from Table A1 and IDs 1 to 60 from Table A2 and IDs 1 to 60 from Table A1 (horizontally flipped).
  - Training set size: 70% of the base dataset.
  - Depth: 10, 30, and 90 frames.
  - Resolutions: 32 × 24, 40 × 30, 80 × 60, and 160 × 120.
  - Epochs: 100.
  - Runs: Three per configuration.
- **Confirmatory experiment C02**:
  - Datasets: IDs 1 to 60 from Table A1 and IDs 1 to 60 from Table A2, IDs 1 to 60 from Table A1 (horizontally flipped) and IDs 1 to 60 from Table A2 (horizontally flipped); IDs 1 to 60 from Table A1 and IDs 1 to 60 from Table A2 and IDs 1 to 60 from Table A1 (horizontally flipped).
  - Training set size: 70% of the base dataset.
  - Depth: 10, 30, and 90 frames.
  - Resolutions: 32 × 24, 40 × 30, 80 × 60, and 160 × 120.
  - Epochs: 100.
  - Runs: Three per configuration.
  - Cross validation: 10 folds.
  - `SBT_balanced_240_70t`

Appendix B. Normal Behavior Samples

Table A1. List of normal behavior samples. Videos were taken from the UCF-Crime dataset [30].

ID	File	Begin	End	ID	File	Begin	End
1	Normal_Videos001_x264.mp4	0:00	0:18	31	Normal_Videos023_x264.mp4	0:00	0:59
2	Normal_Videos002_x264.mp4	0:00	0:55	32	Normal_Videos024_x264.mp4	0:00	0:36
3	Normal_Videos003_x264.mp4	0:00	1:34	33	Normal_Videos029_x264.mp4	0:00	0:29
4	Normal_Videos004_x264.mp4	0:00	0:31	34	Normal_Videos030_x264.mp4	0:00	1:00
5	Normal_Videos005_x264.mp4	0:00	0:13	35	Normal_Videos034_x264.mp4	0:00	0:44
6	Normal_Videos006_x264.mp4	0:00	0:15	36	Normal_Videos036_x264.mp4	0:00	0:44
7	Normal_Videos007_x264.mp4	0:00	0:37	37	Normal_Videos039_x264.mp4	0:00	1:00
8	Normal_Videos008_x264.mp4	0:00	1:26	38	Normal_Videos041_x264.mp4	0:00	0:42
9	Normal_Videos009_x264.mp4	0:08	0:17	39	Normal_Videos043_x264.mp4	0:00	0:58
10	Normal_Videos010_x264.mp4	0:00	0:35	40	Normal_Videos044_x264.mp4	0:00	1:24
11	Normal_Videos011_x264.mp4	0:00	0:30	41	Normal_Videos047_x264.mp4	0:00	1:00
12	Normal_Videos012_x264.mp4	0:00	1:18	42	Normal_Videos048_x264.mp4	0:00	0:56
13	Normal_Videos013_x264.mp4	0:00	0:40	43	Normal_Videos049_x264.mp4	0:00	1:00

Table A1. *Cont.*

ID	File	Begin	End	ID	File	Begin	End
14	Normal_Videos014_x264.mp4	0:00	0:50	44	Normal_Videos051_x264.mp4	0:00	1:19
15	Normal_Videos015_x264.mp4	0:00	0:16	45	Normal_Videos052_x264.mp4	0:00	0:11
16	Normal_Videos017_x264.mp4	0:00	0:28	46	Normal_Videos053_x264.mp4	0:00	0:13
17	Normal_Videos020_x264.mp4	0:00	0:16	47	Normal_Videos054_x264.mp4	0:00	1:06
18	Normal_Videos021_x264.mp4	0:00	1:05	48	Normal_Videos055_x264.mp4	0:00	0:08
19	Normal_Videos022_x264.mp4	0:00	0:13	49	Normal_Videos056_x264.mp4	0:00	0:52
20	Normal_Videos025_x264.mp4	0:00	0:25	50	Normal_Videos057_x264.mp4	0:00	1:00
21	Normal_Videos026_x264.mp4	0:00	1:31	51	Normal_Videos058_x264.mp4	0:00	0:33
22	Normal_Videos027_x264.mp4	0:00	2:44	52	Normal_Videos059_x264.mp4	0:00	1:01
23	Normal_Videos028_x264.mp4	0:00	5:21	53	Normal_Videos061_x264.mp4	0:00	1:00
24	Normal_Videos032_x264.mp4	0:00	0:28	54	Normal_Videos062_x264.mp4	0:00	0:52
25	Normal_Videos033_x264.mp4	0:00	0:56	55	Normal_Videos063_x264.mp4	0:00	0:12
26	Normal_Videos035_x264.mp4	0:04	8:00	56	Normal_Videos064_x264.mp4	0:12	1:10
27	Normal_Videos037_x264.mp4	0:00	0:15	57	Normal_Videos065_x264.mp4	0:00	0:29
28	Normal_Videos038_x264.mp4	0:00	1:39	58	Normal_Videos066_x264.mp4	0:00	0:34
29	Normal_Videos042_x264.mp4	0:00	1:45	59	Normal_Videos067_x264.mp4	0:00	0:36
30	Normal_Videos045_x264.mp4	0:00	0:52	60	Normal_Videos073_x264.mp4	0:08	0:30

Appendix C. Suspicious Behavior Samples

Table A2. List of suspicious behavior samples. Videos were taken from the UCF-Crime dataset [30].

ID	File	Begin	End	ID	File	Begin	End
1	Shoplifting001_x264.mp4	0:00	0:41	31	Shoplifting034_x264.mp4	2:56	3:08
2	Shoplifting005_x264.mp4	0:00	0:25	32	Shoplifting034_x264.mp4	3:12	3:39
3	Shoplifting006_x264.mp4	0:09	0:57	33	Shoplifting034_x264.mp4	3:42	3:43
4	Shoplifting008_x264.mp4	2:10	2:52	34	Shoplifting034_x264.mp4	3:47	4:04
5	Shoplifting009_x264.mp4	0:29	2:26	35	Shoplifting034_x264.mp4	4:09	4:34
6	Shoplifting010_x264.mp4	0:19	0:24	36	Shoplifting036_x264.mp4	0:56	1:44
7	Shoplifting010_x264.mp4	0:43	0:51	37	Shoplifting037_x264.mp4	0:00	0:38
8	Shoplifting012_x264.mp4	1:25	4:26	38	Shoplifting038_x264.mp4	0:50	1:20
9	Shoplifting012_x264.mp4	4:38	5:53	39	Shoplifting039_x264.mp4	0:14	1:10
10	Shoplifting014_x264.mp4	5:51	6:23	40	Shoplifting040_x264.mp4	0:00	0:27
11	Shoplifting014_x264.mp4	6:29	11:43	41	Shoplifting040_x264.mp4	0:34	1:00
12	Shoplifting014_x264.mp4	12:03	18:46	42	Shoplifting040_x264.mp4	1:06	2:24
13	Shoplifting014_x264.mp4	19:01	27:43	43	Shoplifting040_x264.mp4	2:36	4:39
14	Shoplifting015_x264.mp4	0:24	1:07	44	Shoplifting040_x264.mp4	4:50	5:38
15	Shoplifting016_x264.mp4	0:00	0:15	45	Shoplifting040_x264.mp4	5:48	7:12
16	Shoplifting017_x264.mp4	0:00	0:12	46	Shoplifting042_x264.mp4	0:00	1:04
17	Shoplifting018_x264.mp4	0:00	0:14	47	Shoplifting044_x264.mp4	0:00	6:09
18	Shoplifting018_x264.mp4	0:27	0:37	48	Shoplifting047_x264.mp4	0:00	0:32
19	Shoplifting019_x264.mp4	0:06	0:08	49	Shoplifting047_x264.mp4	0:34	0:43
20	Shoplifting020_x264.mp4	1:04	1:17	50	Shoplifting047_x264.mp4	0:47	0:50
21	Shoplifting021_x264.mp4	0:00	1:09	51	Shoplifting047_x264.mp4	0:53	0:59
22	Shoplifting024_x264.mp4	0:00	0:27	52	Shoplifting048_x264.mp4	0:11	0:25
23	Shoplifting025_x264.mp4	0:00	0:56	53	Shoplifting049_x264.mp4	0:00	0:33
24	Shoplifting028_x264.mp4	0:06	0:20	54	Shoplifting051_x264.mp4	0:15	2:32
25	Shoplifting028_x264.mp4	0:23	0:26	55	Shoplifting052_x264.mp4	0:07	0:29
26	Shoplifting029_x264.mp4	0:06	0:27	56	Shoplifting052_x264.mp4	0:34	0:54
27	Shoplifting031_x264.mp4	0:00	0:04	57	Shoplifting052_x264.mp4	1:04	1:29
28	Shoplifting033_x264.mp4	0:00	0:22	58	Shoplifting052_x264.mp4	1:35	2:12
29	Shoplifting034_x264.mp4	0:25	2:36	59	Shoplifting052_x264.mp4	2:16	2:39
30	Shoplifting034_x264.mp4	2:42	2:53	60	Shoplifting053_x264.mp4	0:00	0:43

References

1. Federation, N.R. *2020 National Retail Security Survey*; National Retail Federation: Washington, DC, USA, 2020.
2. Ba, S.O.; Odobez, J. Recognizing Visual Focus of Attention From Head Pose in Natural Meetings. *IEEE Trans. Syst. Man, Cybern. Part B (Cybernetics)* **2009**, *39*, 16–33. [CrossRef]
3. Nayak, N.M.; Sethi, R.J.; Song, B.; Roy-Chowdhury, A.K. Modeling and Recognition of Complex Human Activities. In *Visual Analysis of Humans: Looking at People*; Springer: London, UK, 2011; pp. 289–309. [CrossRef]
4. Rankin, S.; Cohen, N.; Maclennan-Brown, K.; Sage, K. CCTV Operator Performance Benchmarking. In Proceedings of the 2012 IEEE International Carnahan Conference on Security Technology (ICCST), Newton, MA, USA, 15–18 October 2012; pp. 325–330. [CrossRef]
5. DeepCam. Official Website. 2018. Available online: https://deepcamai.com/ (accessed on 6 April 2019).
6. FaceFirst. Official Website. 2019. Available online: https://www.facefirst.com/ (accessed on 6 April 2019).
7. Geng, X.; Li, G.; Ye, Y.; Tu, Y.; Dai, H. Abnormal Behavior Detection for Early Warning of Terrorist Attack. In *AI 2006: Advances in Artificial Intelligence*; Sattar, A., Kang, B.H., Eds.; Springer: Berlin/Heidelberg, Germany, 2006; pp. 1002–1009.
8. Berjon, D.; Cuevas, C.; Moran, F.; Garcia, N. GPU-based implementation of an optimized nonparametric background modeling for real-time moving object detection. *IEEE Trans. Consum. Electron.* **2013**, *59*, 361–369. [CrossRef]
9. Hati, K.K.; Sa, P.K.; Majhi, B. LOBS: Local background subtracter for video surveillance. In Proceedings of the 2012 Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics, Hyderabad, India, 5–7 December 2012; pp. 29–34. [CrossRef]
10. Joshila Grace, L.K.; Reshmi, K. Face recognition in surveillance system. In Proceedings of the 2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, India, 19–20 March 2015; pp. 1–5. [CrossRef]
11. Nurhopipah, A.; Harjoko, A. Motion Detection and Face Recognition for CCTV Surveillance System. *IJCCS (Indones. J. Comput. Cybern. Syst.)* **2018**, *12*, 107. [CrossRef]
12. Hou, L.; Wan, W.; Han, K.; Muhammad, R.; Yang, M. Human detection and tracking over camera networks: A review. In Proceedings of the 2016 International Conference on Audio, Language and Image Processing (ICALIP), Shanghai, China, 11–12 July 2016; pp. 574–580. [CrossRef]
13. Kim, J.S.; Yeom, D.H.; Joo, Y.H. Fast and robust algorithm of tracking multiple moving objects for intelligent video surveillance systems. *IEEE Trans. Consum. Electron.* **2011**, *57*, 1165–1170. [CrossRef]
14. Ling, T.S.; Meng, L.K.; Kuan, L.M.; Kadim, Z.; Baha'a Al-Deen, A.A. Colour-based Object Tracking in Surveillance Application. In Proceedings of the International MultiConference of Engineers and Computer Scientists 2009 (IMECS 2009), Hong Kong, China, 18–20 March 2009; Volume 1.
15. Kang, J.; Kwak, S. Loitering Detection Solution for CCTV Security System. *J. Korea Multimed. Soc.* **2014**, *17*. [CrossRef]
16. Chang, J.Y.; Liao, H.H.; Chen, L.G. Localized detection of abandoned luggage. *EURASIP J. Adv. Signal Process.* **2010**, *2010*, 675784. [CrossRef]
17. Alvar, M.; Torsello, A.; Sanchez-Miralles, A.; Armingol, J.M. Abnormal behavior detection using dominant sets. *Mach. Vis. Appl.* **2014**, *25*, 1351–1368. [CrossRef]
18. Wang, T.; Qiao, M.; Deng, Y.; Zhou, Y.; Wang, H.; Lyu, Q.; Snoussi, H. Abnormal event detection based on analysis of movement information of video sequence. *Opt.-Int. J. Light Electron Opt.* **2018**, *152*, 50–60. [CrossRef]
19. Wu, S.; Wong, H.; Yu, Z. A Bayesian Model for Crowd Escape Behavior Detection. *IEEE Trans. Circuits Syst. Video Technol.* **2014**, *24*, 85–98. [CrossRef]
20. Ouivirach, K.; Gharti, S.; Dailey, M.N. Automatic Suspicious Behavior Detection from a Small Bootstrap Set. In Proceedings of the International Conference on Computer Vision Theory and Applications(VISAPP-2012), Rome, Italy, 24–26 February 2012; pp. 655–658. [CrossRef]
21. Sabokrou, M.; Fathy, M.; Moayed, Z.; Klette, R. Fast and accurate detection and localization of abnormal behavior in crowded scenes. *Mach. Vis. Appl.* **2017**, *28*, 965–985. [CrossRef]
22. Tsushita, H.; Zin, T.T. A Study on Detection of Abnormal Behavior by a Surveillance Camera Image. In *Big Data Analysis and Deep Learning Applications*; Zin, T.T., Lin, J.C.W., Eds.; Springer: Singapore, 2019; pp. 284–291.
23. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.
24. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning Spatiotemporal Features with 3D Convolutional Networks. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 4489–4497. [CrossRef]
25. Intel. Official Website. 2020. Available online: https://software.intel.com/content/www/us/en/develop/tools/openvino-toolkit.html (accessed on 9 February 2020).
26. Hassner, T.; Itcher, Y.; Kliper-Gross, O. Violent flows: Real-time detection of violent crowd behavior. In Proceedings of the 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Providence, RI, USA, 16–21 June 2012; pp. 1–6. [CrossRef]

27. Bermejo Nievas, E.; Deniz Suarez, O.; Bueno García, G.; Sukthankar, R. Violence Detection in Video Using Computer Vision Techniques. In *Computer Analysis of Images and Patterns*; Real, P., Diaz-Pernil, D., Molina-Abril, H., Berciano, A., Kropatsch, W., Eds.; Springer: Berlin/Heidelberg, Germany, 2011; pp. 332–339.
28. Sultani, W.; Chen, C.; Shah, M. Real-World Anomaly Detection in Surveillance Videos. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6479–6488. [CrossRef]
29. Nasaruddin, N.; Muchtar, K.; Afdhal, A.; Dwiyantoro, A.P.J. Deep anomaly detection through visual attention in surveillance videos. *J. Big Data* **2020**, *87*. [CrossRef]
30. University of Central Florida. UCF-Crime Dataset. 2018. Available online: https://webpages.uncc.edu/cchen62/dataset.html (accessed on 23 April 2019).
31. Ishikawa, T.; Zin, T.T. A Study on Detection of Suspicious Persons for Intelligent Monitoring System. In *Big Data Analysis and Deep Learning Applications*; Zin, T.T., Lin, J.C.W., Eds.; Springer: Singapore, 2019; pp. 292–301.
32. Afra, S.; Alhajj, R. Early warning system: From face recognition by surveillance cameras to social media analysis to detecting suspicious people. *Phys. A: Stat. Mech. Appl.* **2020**, *540*, 123151. [CrossRef]
33. Yang, S.; Luo, P.; Loy, C.C.; Tang, X. WIDER FACE: A Face Detection Benchmark. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 5525–5533. [CrossRef]
34. Amos, B.; Ludwiczuk, B.; Satyanarayanan, M. *OpenFace: A General-Purpose Face Recognition Library with Mobile Applications*; Technical Report, CMU-CS-16-118; CMU School of Computer Science: Pittsburgh, PA, USA, 2016.
35. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Hoenix, AZ, USA, 12–17 February 2016.
36. Guo, Y.; Zhang, L.; Hu, Y.; He, X.; Gao, J. MS-Celeb-1M: A Dataset and Benchmark for Large-Scale Face Recognition. In *Computer Vision—ECCV 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 87–102.
37. He, T.; Mao, H.; Yi, Z. Moving object recognition using multi-view three-dimensional convolutional neural networks. *Neural Comput. Appl.* **2017**, *28*, 3827–3835. [CrossRef]
38. Ji, S.; Xu, W.; Yang, M.; Yu, K. 3D Convolutional Neural Networks for Human Action Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 221–231. [CrossRef]
39. Zhang, L.; Zhu, G.; Shen, P.; Song, J. Learning Spatiotemporal Features Using 3DCNN and Convolutional LSTM for Gesture Recognition. In Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), Venice, Italy, 22–29 October 2017; pp. 3120–3128. [CrossRef]
40. Ogwueleka, F.N.; Misra, S.; Colomo-Palacios, R.; Fernandez, L. Neural Network and Classification Approach in Identifying Customer Behavior in the Banking Sector: A Case Study of an International Bank. *Hum. Factors Ergon. Manuf. Serv. Ind.* **2015**, *25*, 28–42. [CrossRef]
41. Cai, X.; Hu, F.; Ding, L. Detecting Abnormal Behavior in Examination Surveillance Video with 3D Convolutional Neural Networks. In Proceedings of the 2016 6th International Conference on Digital Home (ICDH), Guangzhou, China, 2–4 December 2016; pp. 20–24. [CrossRef]
42. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *arXiv* **2015**, arXiv:1512.03385.
43. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
44. Szegedy, C.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1–9. [CrossRef]
45. Varol, G.; Laptev, I.; Schmid, C. Long-Term Temporal Convolutions for Action Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 1510–1517. [CrossRef] [PubMed]
46. Alfaifi, R.; Artoli, A.M. Human Action Prediction with 3D-CNN. *SN Comput. Sci.* **2020**, *1*. [CrossRef]
47. Chandola, V.; Banerjee, A.; Kumar, V. Anomaly Detection: A Survey. *ACM Comput. Surv.* **2009**, *41*. [CrossRef]
48. Jiang, F.; Yuan, J.; Tsaftaris, S.A.; Katsaggelos, A.K. Anomalous Video Event Detection Using Spatiotemporal Context. *Comput. Vis. Image Underst.* **2011**, *115*, 323–333. [CrossRef]
49. Sabokrou, M.; Fathy, M.; Hoseini, M. Video anomaly detection and localisation based on the sparsity and reconstruction error of auto-encoder. *Electron. Lett.* **2016**, *52*, 1122–1124. [CrossRef]
50. Vaswani, N.; Roy Chowdhury, A.; Chellappa, R. Activity recognition using the dynamics of the configuration of interacting objects. In Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Madison, WI, USA, 16–22 June 2003; Volume 2.
51. Ko, K.E.; Sim, K.B. Deep convolutional framework for abnormal behavior detection in a smart surveillance system. *Eng. Appl. Artif. Intell.* **2018**, *67*, 226–234. [CrossRef]
52. Bouma, H.; Vogels, J.; Aarts, O.; Kruszynski, C.; Wijn, R.; Burghouts, G. Behavioral profiling in CCTV cameras by combining multiple subtle suspicious observations of different surveillance operators. In *Signal Processing, Sensor Fusion, and Target Recognition XXII*; Kadar, I., Ed.; International Society for Optics and Photonics, SPIE: San Diego, CA, USA, 2013; Volume 8745, pp. 436–444. [CrossRef]
53. Koller, C.I.; Wetter, O.E.; Hofer, F. 'Who's the Thief?' The Influence of Knowledge and Experience on Early Detection of Criminal Intentions. *Appl. Cogn. Psychol.* **2016**, *30*, 178–187. [CrossRef]

54. Grant, D.; Williams, D. The importance of perceiving social contexts when predicting crime and antisocial behaviour in CCTV images. *Leg. Criminol. Psychol.* **2011**, *16*, 307–322. [CrossRef]
55. Koller, C.I.; Wetter, O.E.; Hofer, F. What Is Suspicious When Trying to be Inconspicuous? Criminal Intentions Inferred From Nonverbal Behavioral Cues. *Perception* **2015**, *44*, 679–708. [CrossRef]
56. Troscianko, T.; Holmes, A.; Stillman, J.; Mirmehdi, M.; Wright, D.; Wilson, A. What happens next? The predictability of natural behaviour viewed through CCTV cameras. *Perception* **2004**, *33*, 87–101. [CrossRef]
57. Altemir, V. La comunicación no verbal como herramienta en la videovigilancia. In *Comportamiento no Verbal: Más Allá de la Cmunicación y el Lenguaje*; Pirámide: Madrid, Spain, 2016; pp. 225–228.
58. Kim, H.; Jeong, Y.S. Sentiment Classification Using Convolutional Neural Networks. *Appl. Sci.* **2019**, *9*, 2347. [CrossRef]
59. Roth, H.R.; Yao, J.; Lu, L.; Stieger, J.; Burns, J.E.; Summers, R.M. Detection of Sclerotic Spine Metastases via Random Aggregation of Deep Convolutional Neural Network Classifications. In *Recent Advances in Computational Methods and Clinical Applications for Spine Imaging*; Yao, J., Glocker, B., Klinder, T., Li, S., Eds.; Springer International Publishing: Cham, Switzerland, 2015; pp. 3–12. [CrossRef]
60. Fujimoto Lab. 3D Convolutional Neural Network for Video Classification, Code Repository. 2017. Available online: https://github.com/kcct-fujimotolab/3DCNN (accessed on 28 April 2019).
61. Google. Google Colaboratory. 2017. Available online: https://colab.research.google.com/ (accessed on 29 April 2019).